Python les-materialen

# Opdrachten datastructuren met oplossingen

## Test je eigen kennis.

\*\* Beantwoord de volgende vragen \*\*

Write a brief description of all the following Object Types and Data Structures we’ve learned about:

## Numbers

Write an equation that uses multiplication, division, an exponent, addition, and subtraction that is equal to 100.25.

Hint: This is just to test your memory of the basic arithmetic commands, work backwards from 100.25

# Your answer is probably different  
(60 + (10 \*\* 2) / 4 \* 7) - 134.75

100.25

Answer these 3 questions without typing code. Then type code to check your answer.

What is the value of the expression 4 \* (6 + 5)  
  
What is the value of the expression 4 \* 6 + 5   
  
What is the value of the expression 4 + 6 \* 5

4 \* (6 + 5)

44

4 \* 6 + 5

29

4 + 6 \* 5

34

What is the *type* of the result of the expression 3 + 1.5 + 4?

**Answer: Floating Point Number**

What would you use to find a number’s square root, as well as its square?

# Square root:  
100 \*\* 0.5

10.0

# Square:  
10 \*\* 2

100

## Strings

Given the string ‘hello’ give an index command that returns ‘e’. Enter your code in the cell below:

s = 'hello'  
# Print out 'e' using indexing  
  
s[1]

'e'

Reverse the string ‘hello’ using slicing:

s ='hello'  
# Reverse the string using slicing  
  
s[::-1]

'olleh'

Given the string ‘hello’, give two methods of producing the letter ‘o’ using indexing.

s ='hello'  
# Print out the 'o'  
  
# Method 1:  
  
s[-1]

'o'

# Method 2:  
  
s[4]

'o'

## Lists

Build this list [0,0,0] two separate ways.

# Method 1:  
[0]\*3

[0, 0, 0]

# Method 2:  
list2 = [0,0,0]  
list2

[0, 0, 0]

Reassign ‘hello’ in this nested list to say ‘goodbye’ instead:

list3 = [1,2,[3,4,'hello']]

list3[2][2] = 'goodbye'

list3

[1, 2, [3, 4, 'goodbye']]

Sort the list below:

list4 = [5,3,4,6,1]

# Method 1:  
sorted(list4)

[1, 3, 4, 5, 6]

# Method 2:  
list4.sort()  
list4

[1, 3, 4, 5, 6]

## Dictionaries

Using keys and indexing, grab the ‘hello’ from the following dictionaries:

d = {'simple\_key':'hello'}  
# Grab 'hello'  
  
d['simple\_key']

'hello'

d = {'k1':{'k2':'hello'}}  
# Grab 'hello'  
  
d['k1']['k2']

'hello'

# Getting a little tricker  
d = {'k1':[{'nest\_key':['this is deep',['hello']]}]}

# This was harder than I expected...  
d['k1'][0]['nest\_key'][1][0]

'hello'

# This will be hard and annoying!  
d = {'k1':[1,2,{'k2':['this is tricky',{'tough':[1,2,['hello']]}]}]}

# Phew!  
d['k1'][2]['k2'][1]['tough'][2][0]

'hello'

Can you sort a dictionary? Why or why not?

**Answer: No! Because normal dictionaries are *mappings* not a sequence.**

## Tuples

What is the major difference between tuples and lists?

**Tuples are immutable!**

How do you create a tuple?

t = (1,2,3)

## Sets

What is unique about a set?

**Answer: They don’t allow for duplicate items!**

Use a set to find the unique values of the list below:

list5 = [1,2,2,33,4,4,11,22,3,3,2]

set(list5)

{1, 2, 3, 4, 11, 22, 33}

## Booleans

For the following quiz questions, we will get a preview of comparison operators. In the table below, a=3 and b=4.

Operator

Description

Example

==

If the values of two operands are equal, then the condition becomes true.

(a == b) is not true.

!=

If values of two operands are not equal, then condition becomes true.

(a != b) is true.

>

If the value of left operand is greater than the value of right operand, then condition becomes true.

(a > b) is not true.

<

If the value of left operand is less than the value of right operand, then condition becomes true.

(a < b) is true.

>=

If the value of left operand is greater than or equal to the value of right operand, then condition becomes true.

(a >= b) is not true.

<=

If the value of left operand is less than or equal to the value of right operand, then condition becomes true.

(a <= b) is true.

What will be the resulting Boolean of the following pieces of code (answer fist then check by typing it in!)

# Answer before running cell  
2 > 3

False

# Answer before running cell  
3 <= 2

False

# Answer before running cell  
3 == 2.0

False

# Answer before running cell  
3.0 == 3

True

# Answer before running cell  
4\*\*0.5 != 2

False

Final Question: What is the boolean output of the cell block below?

# two nested lists  
l\_one = [1,2,[3,4]]  
l\_two = [1,2,{'k1':4}]  
  
# True or False?  
l\_one[2][0] >= l\_two[2]['k1']

False

## Great Job on your first assessment!